ARL-TN-0710 ● Nov 2015

US Army Research Laboratory

Dependency Tree Annotation Software

by Rhea Dedhia

NOTICES

Disclaimers

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of manufacturer's or trade names does not constitute an official endorsement or approval of the use thereof.

Destroy this report when it is no longer needed. Do not return it to the originator.

ARL-TN-0710 ● Nov 2015

US Army Research Laboratory

Dependency Tree Annotation Software

by Rhea Dedhia
Computational and Information Sciences Directorate, ARL

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED (From - To)
November 2015	Final	06/2015–08/2015

4. TITLE AND SUBTITLE
Dependency Tree Annotation Software

5a. CONTRACT NUMBER
SEAP-ASEE

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Rhea Dedhia

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
US Army Research Laboratory
ATTN: RDRL-CII-T
2800 Powder Mill Road
Adelphi, MD 20783-1138

8. PERFORMING ORGANIZATION REPORT NUMBER
ARL-TN-0710

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT
This report presents Dependency Tree Editor (DTE), a software system built as part of a Science and Engineering Apprentice Program (SEAP) summer internship. The system supports interactive visualization and editing of dependency trees—a formalism frequently used in computational linguistics to represent the syntactic structure of sentences. DTE enables users to annotate words with their parts of speech and create, label, and delete dependency links between words. DTE supports the widely used Conference on Computational Natural Language Learning (CoNLL)-X format as well as several other file formats, and it provides numerous options for customizing how dependency trees are displayed. Built entirely in Java, it can run on a wide range of platforms.

15. SUBJECT TERMS
dependency tree, software

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Stephen C Tratz
Unclassified	Unclassified	Unclassified	UU	18	19b. TELEPHONE NUMBER (Include area code) 301-394-2305

Contents

List of Figures

1. Background and Introduction

A dependency tree maps the syntactic dependency relationships between the parts of a sentence. Dependency parsers use the syntactic patterns present in a language in order to generate dependency trees for sentences. These can be used to understand the meaning of a sentence and/or extract information from it. However, the dependency trees automatically generated from long and complex sentences tend to have mistakes. As a result, linguists often have to manually fix them. Since there are few publicly available tools to edit dependency trees in an intuitive, efficient manner, this can be time consuming. Additionally, editing dependency trees using text editors is highly error prone.

In order to help resolve these issues, I have created a software application called Dependency Tree Editor (DTE) that can read files in Computational Natural Language Learning (CoNLL)-X format and use them to render dependency trees that can be easily edited via a graphical interface. Once the user has finished editing the tree, it can be saved to disk as a CoNLL-X file.

The CoNLL-X file format was created by the Conference on CoNLL and has become a popular format for storing dependency trees. CoNLL is organized by the Association for Computational Linguistics (ACL) Special Interest Group on Natural Language Learning (SIGNLL).

2. Features Description

As shown in Fig. 1, words are represented as nodes and the relationships between them are represented as edges. The edges can be labeled with a dependency relation label.

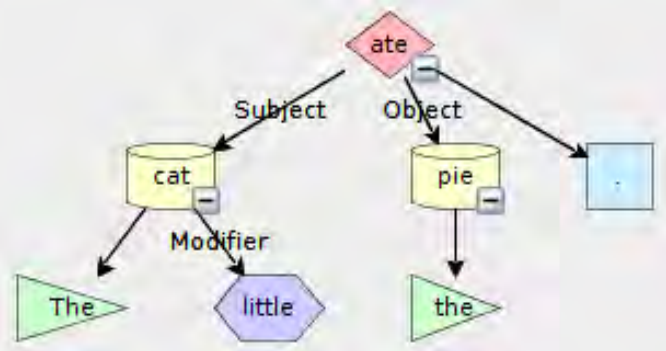

Fig. 1 Manually created dependency tree for the sentence, "The little cat ate the pie"

The user can easily assign shapes and colors to specific part-of-speech (POS) tags. In Fig. 1, verbs are shown as red rhombuses, nouns as yellow cylinders,

adjectives as purple hexagons, determiners as green triangles, and punctuation as blue rectangles. The display is automatically updated to reflect any changes in these settings.

The DTE graphical user interface consists of 5 main components, which are illustrated in Fig. 2. The sentence bar on the left allows the user to easily switch between sentences and their associated dependency trees. Sentences can be added to the sentence bar by opening a CoNLL-X file, typing sentences into a text field, or opening a .txt file containing 1 sentence per line. The user can switch to another dependency tree by clicking on a different sentence in the sentence bar. Relationships between words can be easily created and removed, and users can customize how they want to be able to do this by selecting or deselecting options in the menu bar.

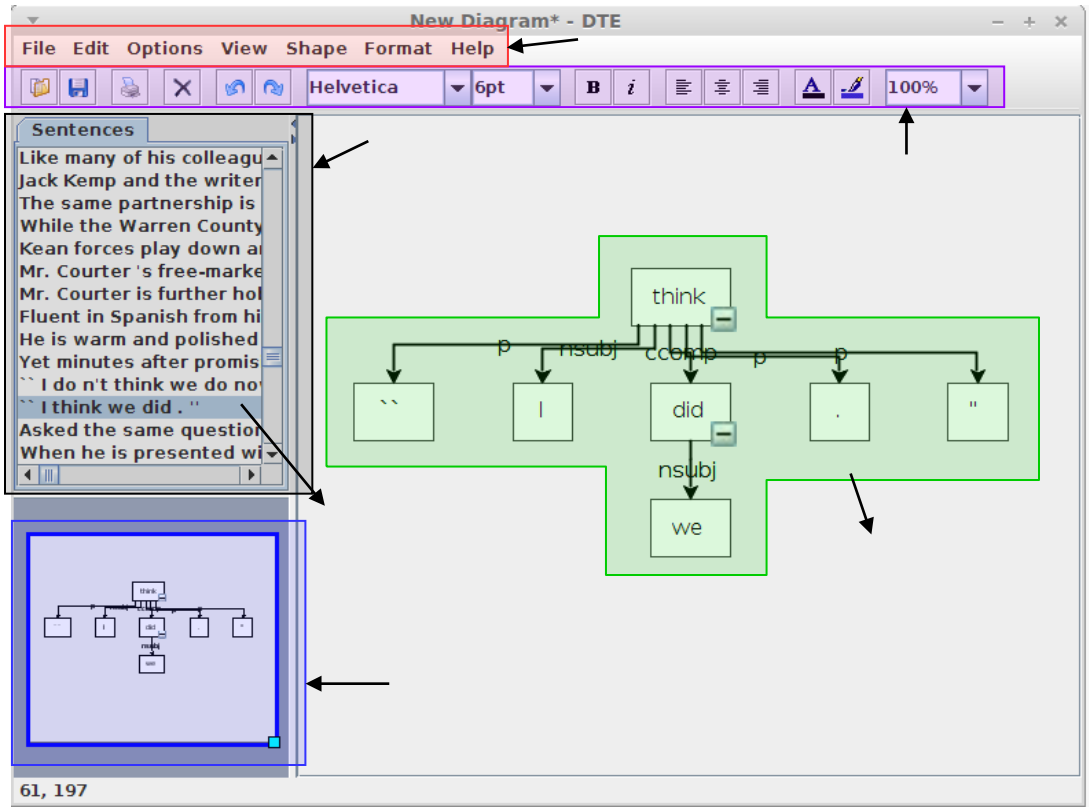

Fig. 2 DTE after a .conll file has been opened and a sentence has been clicked on

The user can easily set a node's POS tag (e.g., Noun, Verb) and an edge's dependency label by selecting the appropriate label from the submenu containing the corresponding list. The user can easily add to these lists as well. This lets users customize the POS tags and dependency labels that they want to use in their dependency trees. In addition, when a user opens a CoNLL-X file, all the POS tags and dependency labels in that file are added to the program's lists of POS

tags and dependency labels. Users may also change the dependency label for a particular edge by double clicking on it and selecting the appropriate label from a drop-down list. As shown in Fig. 3, hovering over a node brings up a tool tip with that node's POS tag.

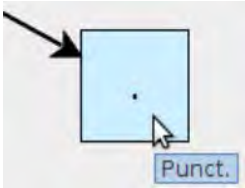

Fig. 3 Hovering over a node brings up a tool tip with that node's POS tag

The user can also adjust the font, font color, font size, arrow color, arrow thickness, and label positioning. The user can zoom in or zoom out using the menu bar, the icon menu bar, or the outline (see Fig. 2), or by pressing CTRL and using the ± keys or the mouse scrollwheel.

Other aspects of the user interface are customizable as well. Users can decide whether they want the program to automatically format their dependency tree and whether they want their arrows to be straight (see Fig. 1) or have right angles (see Fig. 2). Users can also choose whether or not they want the outline (see Fig. 2) to appear. The created dependency tree can be saved as an image, .mxe (a mxGraph editing file), a .conll file, and several other file formats.

DTE uses the open source Java version of the diagramming library mxGraph. (Download link: https://github.com/jgraph/jgraphx).

3. Other Software

Tree Editor (TrEd) (https://ufal.mff.cuni.cz/tred/) is a publicly available, customizable, and programmable graphical editor and viewer for tree-like structures. It is useful for professional annotators but has a steep learning curve and can be unintuitive to new users. DTE is designed to be accessible so that new users can quickly start using it to create and edit dependency trees.

4. Conclusion

With DTE, linguists can easily load dependency trees from .conll files and edit them. They can also input unannotated sentences and easily create trees for them, which can then be saved as .conll files, a frequently used format for dependency trees, or in one of the many file types that mxGraph supports.

INTENTIONALLY LEFT BLANK.

Appendix. Documentation of Software Features

A-1 Menu Bar

See Fig. 2.

- File
  - "Open File" allows the user to select and open a .mxe file (standard mxGraph editing file; extended markup language [XML] format), PNG+XML (.png) file or .vdx file (XML drawing file) from their files.*
  - "Open .conll File" allows the user to select and open a .conll file from existing files. It then draws the parent-child relationships depicted in the opened .conll file.
  - "Open .sconll File" opens a special version of the .conll file in order to allow the user to deal with the individual morphemes of a token separately and give them part-of-speech (POS) tags (see *"Split Word"*).
  - "Save" saves the file that is currently open. If the user opens a .conll file, it will not be saved when "Save" is clicked. To do this, the user must use either "Save As" or "Save As .conll File." *
  - "Save As" allows the user to save the dependency tree as a PNG+XML file (.png), a .mxe file , a .txt file, a .svg file, a VML file (.html), an HTML file (.html), a .bmp file, a .wbmp file, a .jpg file, a .jpeg file, or a .gif file.*
  - "Save As .conll File" allows users to save the dependency trees that they have edited or created as a .conll file.
  - "Print" prints the dependency tree currently being displayed. *
  - "Exit" closes the program.*
- Edit
  - "Delete" deletes any selected edges if "allow edge deletion" is selected.
  - "Split Word" opens a dialogue box that allows the user to place a space or spaces in the word where the user wants the word to be split. This can be used to distinguish between the individual morphemes of a

*Features already implemented by mxGraph.

token and is particularly of value for languages such as Arabic, which can combine several words into a single token. Once this has been clicked, the file will automatically be saved as a .sconll file.

- "Unsplit word" recombines split words. If the user splits a word or words and recombines all of them, the file will automatically be saved as a standard .conll file.
- "Add Arrow Tags With File" and "Add POS Tags With File" let the user open a file with dependency labels or POS tags, which will be added to the program's stored label lists.
- "Set Arrow Tags With File" and "Set POS Tags With File" (Fig. A-1) clears the program's internal lists of dependency labels and POS tags and then adds the ones in the file to that list.
- "Type New Arrow Tag" and "Type New POS Tag" let the user enter new dependency labels and POS tags.
- "Add Sentences With File" lets the user open a file that contains sentences that will be added to the sentence bar on the left side of the window (see Fig. 2).
    - Sentence Format: Each line will be added as an individual sentence. Punctuation at the end of each sentence, quotation marks, colons, and semicolons will become separate nodes. Unlike *Paragraph Format* (Fig. A-1), this supports abbreviations with periods.

Fig. A-1 (Left) When one clicks "Split Word," a box pops up that asks the user to enter a space or spaces where the user wants to split the string; (middle) the word with spaces in it where it will be split; and (right) the split word

- Additional Options (enable or disable):
    - "Split Verb Contractions"
    - "Split $ and %" – splits $ and % symbols from any numbers they are attached to.
    - For example, if both options are selected, the sentence *"They're going to buy $30 worth of items with a 10% sales tax" he said!* will be

parsed into the following tokens: *“ They ‘re going to buy $ 30 worth of items with a 10 % sales tax ” he said !*

- Node Format: Each line will be added as a word/word part, with blank lines to distinguish between sentences.
- Paragraph Format: The paragraph is separated into sentences with punctuation as an indicator of the end of a sentence. See *Sentence Format* and *Additional Options* above for information as to how these sentences are divided up into nodes. This feature currently does not support abbreviations with punctuation, such as *Mr.* and *U.S.A.*

- “Type New Sentence” lets the user type in a sentence that will be added to the sentence bar. (See *Sentence Format* and *Additional Options* under “Add Sentences with File” to see how sentences are divided up into nodes.)
- “Format” formats the dependency tree that the user has created.

- Options
  - “Settings” lets the user associate 1 color and 1 shape with each POS tag in the program’s list of POS tags. The default color is white and the default shape is a rectangle.
  - When “Automatically Format” is selected, the tree that the user has created will format whenever the user moves a cell or cells or releases the mouse. (Default is not selected.)
  - When “Move Cells On Mouse Drag” is selected, all the cells in graph will be moved with the mouse whenever the user drags the mouse. When it is not selected, dragging the mouse will select the area that the user dragged the mouse over. (Default is selected.)
  - When “Connect On Overlap” is selected, when the user drags a node near another node that isn’t a child of that node, the node underneath will be highlighted. If the user drops the node while another node is highlighted, an edge will be drawn from the highlighted node to the node that was dragged over it. (Default is selected.)
  - When “Drag and Drop Arrows” is selected, the user can hover the mouse over a node until its outer edge is highlighted in green, then click and drag an arrow from it to another node. (Default is selected.)

- When “Delete Arrows by Pulling off of Tree” is selected, if the user drags an edge away from its source and target nodes, it will be deleted. Otherwise, the user will not be able to move edges. (Default is selected.)
- When “Allow Arrow Deletion” is selected, the user is able to delete edges by selecting 1 or more edges and clicking the delete toolbar button or by clicking the “delete” option in the right click drop-down menu or in the edit menu item in the menu bar. Deselecting “Allow Arrow Deletion” causes “Delete Arrows by Pulling off of Tree” to be deselected and disables it until “Allow Arrow Deletion is reselected. (Default is selected.)
- When “Format With Right Angle Arrows” is selected, whenever the graph is formatted, the arrows will be redrawn with right angles (see Fig. 2). Otherwise, arrows will be drawn as straight lines (see Fig. 1). (Default is selected.)
- When “Double Click To Expand/Collapse Node” is selected, the user can double click on a node to collapse or expand all its children. *Note: When a node’s children are collapsed, its border will become thicker and it will become darker.* (Default is selected.)
- When “Click Plus/Minus Icon to Expand/Collapse Node” is selected, a plus (+) or minus (–) icon will appear at the bottom of each node that has 1 or more children. This icon can be clicked to expand or collapse the node’s children. (Default is selected.)

- View
  - “Zoom” lets the user select a value to zoom the graph to. The user can also select “Custom” and enter in a customized zoom percentage.*
  - “Zoom in” and “Zoom out” cause the graph to zoom in or zoom out by a specific amount. *
  - When “Outline” is selected, a small version of the graph and the dependency tree will appear in a box in the bottom-left corner of window (see Fig. 2). The user can increase or decrease the field of view using this frame. (Default is selected.) *
- Shape
  - “To Back” moves an edge or node to the back (z-order). *
  - “To Front” moves an edge or node to the front (z-order). *

- Format contains options that let the user format the arrow label and node text (font, color, background color, position) and the padding between the edges and the nodes. *
- Help gives the user information about mxGraph.*

A-2 Icon Menu Bar*

See Fig. 2.

- Open Icon – “Menu Bar” => “File” => “Open File”
- Save Icon – “Menu Bar” => “File” => “Save File”
- Print Icon – “Menu Bar” => “File” => “Print”
- Delete Icon – “Menu Bar” => “Edit” => “Delete”
- Undo Icon/Redo Icon
- Icons that edit selected edges and nodes
    - Font Drop-Down Menu – changes the font of the text.
    - Font Size Drop-Down Menu – changes the size of the text font.
    - Bold/Italic Icons – makes the text bold and/or italic.
    - Left Align/Center Align/Right Align Icons – aligns the text.
    - Font Color Icon – changes the color of the text.
    - Line Color Icon – changes the color of the edges and the color of the borders of the nodes.
- Zoom Drop-Down Menu – lets the user choose an amount to zoom by from a list.

A-3 Drop-Down Menu

Generated on right click; see Fig. 2.

- “Delete” – “Menu Bar” => “Edit” => “Delete” (Enabled if only edges are selected and allow edge deletion is true.)
- “Select Vertices” selects all nodes. *
- “Select Edges” selects all edges. *
- “Select All” selects all nodes and edges. *

- "Set POS Tag" (Fig. A-2) lets the user set the POS tag of the selected node to a POS tag from the program's list of POS Tags. (Enabled if only 1 node is selected.)

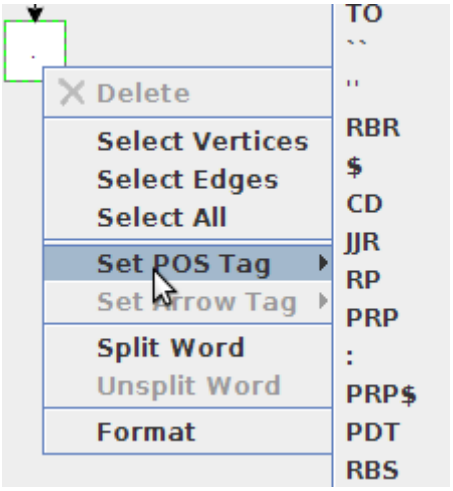

Fig. A-2 To set the POS tag of a node, right click on the node, select "Set POS Tag," and choose a POS tag from the list displayed

- "Set Arrow Tag" lets the user choose the edge's dependency label from the program's stored list of dependency labels. (Enabled if 1 edge is selected.)
- "Split Word" (see *"Split Word"* under *"Edit"*) (Enabled if 1 node is selected and it is not already a word part.)
- "Unsplit Word" (see *"Unsplit Word"* under *"Edit"*) (Enabled if 1 split word is selected.)
- "Format" (see *"Format"* under *"Edit"*)

A-4 Switch Between Dependency Trees

The user can switch between dependency trees by selecting a different sentence in the sentence bar (see Fig. 2).

1 DEFENSE TECH INFO CTR
(PDF) DTIC OCA

2 US ARMY RSRCH LAB
(PDF) IMAL HRA MAIL & RECORDS MGMT
RDRL CIO LL TECHL LIB

1 GOVT PRINTG OFC
(PDF) A MALHOTRA

1 US ARMY RSRCH LAB
(PDF) RDRL CII T
S TRATZ